

INFORMATION PROCESSING APPARATUS AND PRINTING SYSTEM AND MODULE DECIDING METHOD

Patent Number: JP2002108580
Publication date: 2002-04-12
Inventor(s): MIYATA JUNICHI
Applicant(s): CANON INC
Requested Patent: ☐ JP2002108580
Application Number: JP20000292930 20000926
Priority Number(s):
IPC Classification: G06F3/12; B41J29/00; B41J29/38
EC Classification:
Equivalents:

Abstract

PROBLEM TO BE SOLVED: To provide an information processing apparatus, a printing system, and a module deciding method capable of reducing redundancy, reducing memory shortage, simplifying printing performance, and improving a printing speed by deciding a module group to be integrated into a driver module which controls output equipment according to a printing function set by a user.

SOLUTION: A user is allowed to set his or her desired printing function in printer equipment having a printing function by an UI part 402, and a module group 403 to be integrated into a driver which controls the printer equipment is decided so that when the printing function set by the user is, for example, aimed at watermark printing, a module 1 can be loaded, or when the printing function is aimed at page printing, a module 2 can be loaded, and a printer equipment control command is generated by the decided module group, and outputted to the printer equipment.

Data supplied from the esp@cenet database - I2

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2002-108580
(P2002-108580A)

(43) 公開日 平成14年4月12日 (2002. 4. 12)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード(参考)
G 0 6 F 3/12		G 0 6 F 3/12	C 2 C 0 6 1
B 4 1 J 29/00		B 4 1 J 29/38	Z 5 B 0 2 1
29/38		29/00	Z

審査請求 未請求 請求項の数 8 O L (全 15 頁)

(21) 出願番号 特願2000-292930(P2000-292930)

(22) 出願日 平成12年9月26日 (2000. 9. 26)

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 宮田 淳一

東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(74) 代理人 100076428

弁理士 大塚 康德 (外2名)

Fターム(参考) 2C061 AP01 CL10 HH03 HJ08 HK04

HN05 HN15

5B021 AA01 BB01 BB04 BB10 CC02

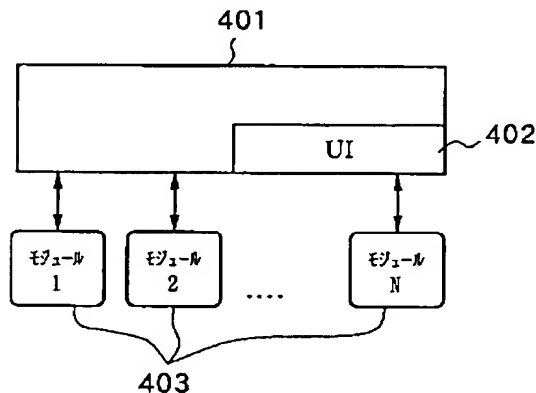
CC05

(54) 【発明の名称】 情報処理装置、印刷システム、及びモジュール決定方法

(57) 【要約】

【課題】 ユーザが設定した印刷機能に応じて、出力機器を制御するドライバモジュールに組み込むモジュール群を決定することにより、冗長性を減少させ、メモリ不足を軽減でき、印刷処理を簡潔化し、印刷速度を向上させた情報処理装置、印刷システム、及びモジュール決定方法を提供する。

【解決手段】 UI部402にて印刷機能を有するプリンタ機器に対してユーザに所望の印刷機能を設定させ、ユーザが設定した印刷機能が、例えば透かし印刷機能の場合にはモジュール1をロードし、ページ印刷機能の場合にはモジュール2をロードするというように、プリンタ機器を制御するドライバに組み込むべきモジュール群403を決定し、決定されたモジュール群によりプリンタ機器制御コマンドを生成してプリンタ機器に出力する。



【特許請求の範囲】

【請求項1】 印刷機能を有する出力機器と接続され、前記出力機器を制御する出力機器制御コマンドを生成して出力する情報処理装置であって、印刷機能を有する出力機器に対してユーザが所望の印刷機能を設定するための設定手段と、設定された印刷機能に応じて、前記出力機器を制御するドライバモジュールに組み込むモジュール群を前記ドライバモジュール初期化時に決定する決定手段と、決定されたモジュール群により出力機器制御コマンドを生成し、前記出力機器に出力する手段とを有することを特徴とする情報処理装置。

【請求項2】 前記設定手段は、前記ドライバモジュールが表示するユーザインタフェースによりユーザに印刷機能を設定させることを特徴とする請求項1に記載の情報処理装置。

【請求項3】 前記モジュール群は、前記設定された印刷機能により個々に分割されたサブモジュールで構成されていることを特徴とする請求項1に記載の情報処理装置。

【請求項4】 前記決定手段は、前記設定された印刷機能が透かし印刷機能の場合に、当該印刷機能に対応するサブモジュールの組み込みを前記ドライバモジュール初期化時に決定することを特徴とする請求項3に記載の情報処理装置。

【請求項5】 前記決定手段は、前記設定された印刷機能がページ印刷機能の場合に、当該印刷機能に対応するサブモジュールの組み込みを前記ドライバモジュール初期化時に決定することを特徴とする請求項3に記載の情報処理装置。

【請求項6】 請求項1乃至請求項5の何れか一項に記載の情報処理装置と印刷機能を有する出力機器とで構成されることを特徴とする印刷システム。

【請求項7】 印刷機能を有する出力機器と接続され、前記出力機器を制御する出力機器制御コマンドを生成して出力する情報処理装置におけるモジュール決定方法であって、印刷機能を有する出力機器に対してユーザが所望の印刷機能を設定するための設定工程と、設定された印刷機能に応じて、前記出力機器を制御するドライバモジュールに組み込むモジュール群を前記ドライバモジュール初期化時に決定する決定工程と、決定されたモジュール群により出力機器制御コマンドを生成し、前記出力機器に出力する工程とを有することを特徴とするモジュール決定方法。

【請求項8】 印刷機能を有する出力機器と接続され、前記出力機器を制御する出力機器制御コマンドを生成して出力する情報処理装置におけるモジュール決定方法のプログラムが格納されたコンピュータ読み取り可能な記憶媒体であって、

印刷機能を有する出力機器に対してユーザが所望の印刷機能を設定するためのプログラムと、設定された印刷機能に応じて、前記出力機器を制御するドライバモジュールに組み込むモジュール群を前記ドライバモジュール初期化時に決定するプログラムと、決定されたモジュール群により出力機器制御コマンドを生成し、前記出力機器に出力するプログラムとを含むことを特徴とする記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、印刷機能を有する出力機器と接続され、前記出力機器を制御する出力機器制御コマンドを生成して出力する情報処理装置、印刷システム、及びモジュール決定方法に関するものである。

【0002】

【従来の技術】一般的に、情報処理装置で動作するオペレーティングシステム（OS）上の印刷システムにおいて、アプリケーションプログラムからの印刷データは論理描画要求として、プリンタ機器を制御するプリンタ機器制御プログラム（以下、プリンタドライバ）へと送信される。このプリンタドライバは論理描画要求を解釈し、プリンタ機器が認識するプリンタ制御コマンドを生成する。通常、このような機能を遂行するプリンタドライバは複数のプログラムモジュールで構成されている。これらはオペレーティングシステムが規定したオペレーティングシステムとプリンタドライバとの間のインタフェースを実装するモジュールであり、存在するインタフェース分のモジュールがプリンタドライバ内には存在する。

【0003】通常、モジュールはオペレーティングシステムの定める一つのインタフェースに対して一つ存在する。

【0004】

【発明が解決しようとする課題】しかしながら、上記従来例では、オペレーティングシステムにより定められた、オペレーティングシステムとプリンタドライバとのインタフェースを提供するプリンタドライバモジュールは、プリンタドライバ機能が豊富になるにつれて、該モジュール内部の処理手順やモジュールサイズが肥大化する。これらを1つのモジュールとして扱うことはプリンタドライバ機能の中で印刷時に不必要な冗長部分をも同様に扱うこととなるため非効率である。また、冗長部分も同一として1つのモジュールとして扱うことは、モジュールをロード時のファイルサイズの肥大化によるメモリ不足や印刷速度の低下を引き起こす恐れがある。

【0005】実際に、プリンタドライバで必要となる機能は、プリンタドライバが提供するユーザインタフェース上からユーザが設定した機能を印刷する機能のみであった。

【0006】本発明は、上記課題を解決するためになさ

れたもので、ユーザが設定した印刷機能に応じて、出力機器を制御するドライバモジュールに組み込むモジュール群を前記ドライバモジュール初期化時に決定することにより、冗長性を減少させ、メモリ不足を軽減でき、印刷処理を簡潔化し、印刷速度を向上させた情報処理装置、印刷システム、及びモジュール決定方法を提供することを目的とする。

【0007】

【課題を解決するための手段】上記目的を達成するために、本発明は、印刷機能を有する出力機器と接続され、前記出力機器を制御する出力機器制御コマンドを生成して出力する情報処理装置であって、印刷機能を有する出力機器に対してユーザが所望の印刷機能を設定するための設定手段と、設定された印刷機能に応じて、前記出力機器を制御するドライバモジュールに組み込むモジュール群を前記ドライバモジュール初期化時に決定する決定手段と、決定されたモジュール群により出力機器制御コマンドを生成し、前記出力機器に出力する手段とを有することを特徴とする。

【0008】また上記目的を達成するために、本発明は、印刷機能を有する出力機器と接続され、前記出力機器を制御する出力機器制御コマンドを生成して出力する情報処理装置におけるモジュール決定方法であって、印刷機能を有する出力機器に対してユーザが所望の印刷機能を設定するための設定工程と、設定された印刷機能に応じて、前記出力機器を制御するドライバモジュールに組み込むモジュール群を前記ドライバモジュール初期化時に決定する決定工程と、決定されたモジュール群により出力機器制御コマンドを生成し、前記出力機器に出力する工程とを有することを特徴とする。

【0009】

【発明の実施の形態】以下、図面を参照しながら本発明に係る実施の形態を詳細に説明する。

【0010】図1は、本実施形態による印刷システムの構成を示すブロック図である。尚、本発明の機能が実行されるのであれば、単体の機器であっても、複数の機器からなるシステムであっても、LAN、WAN等のネットワークを介して接続され、所定の処理が行われるシステムであっても本発明を適用できることは言うまでもない。

【0011】同図において、3000はホストコンピュータであり、ROM3のプログラムROM3b或いは外部メモリ11に記憶された文書処理プログラム等に基づいて図形、イメージ、文字、表（表計算等を含む）等が混在した文書処理を実行するCPU1を備え、システムバス4に接続される各デバイスをCPU1が総括的に制御する。

【0012】尚、このROM3のプログラムROM3b或いは外部メモリ11には、CPU1の制御プログラムであるオペレーティングシステムプログラム（以下、O

S）等が記憶され、またROM3のフォントROM3a或いは外部メモリ11には、上述の文書処理の際に使用するフォントデータ等が記憶され、同様にROM3のデータ用ROM3c或いは外部メモリ11には上述の文書処理等を行う際に使用する各種データが記憶されている。

【0013】2はRAMであり、CPU1の主メモリ、ワークエリア等として機能する。5はキーボードコントローラ（KBC）であり、キーボード9や不図示のポインティングデバイスからのキー入力を制御する。6はCRTコントローラ（CRTC）であり、CRTディスプレイ（CRT）10への表示を制御する。7はディスクコントローラ（DKC）であり、ブートプログラム、種々のアプリケーション、フォントデータ、ユーザファイル、編集ファイル等を記憶するハードディスク（HD）、フロッピー（登録商標）ディスク（FD）等の外部メモリ11へのアクセスを制御する。8はプリンタコントローラ（PRTC）であり、所定の双方向性インタフェース（インタフェース）21を介してプリンタ1500に接続され、プリンタ1500との通信制御処理を実行する。

【0014】尚、CPU1は、例えばRAM2上に設定された表示情報RAMへのアウトラインフォントの展開（ラスターライズ）処理を実行し、CRT10上でのWYSIWYGを可能としている。また、CPU1は、CRT10上の不図示のマウスカーソル等で指示されたコマンドに基づいて登録された種々のウインドウを開き、種々のデータ処理を実行する。ユーザは印刷を実行する際、印刷の設定に関するウインドウを開き、印刷設定や、印刷シーケンスの選択を含むプリンタドライバに対する印刷処理方法の設定を行える。

【0015】一方、プリンタ1500において、12はプリンタCPUであり、ROM13のプログラムROM13bに記憶された制御プログラム等或いは外部メモリ14に記憶された制御プログラム等に基づいてシステムバス15に接続された各種のデバイスへのアクセスを総括的に制御し、印刷部インタフェース（I/F）16を介して接続された印刷部（プリンタエンジン）17に出力情報としての画像信号を出力する。

【0016】尚、このROM13のプログラムROM13bには、CPU12の制御プログラム等が記憶され、またROM13のフォントROM13aには、上述の出力情報を生成する際に使用するフォントデータ等が記憶され、同様にROM13のデータROM13cにはハードディスク等の外部メモリ14がないプリンタの場合、ホストコンピュータ3000上で利用される情報等が記憶されている。

【0017】また、CPU12は入力部18を介してホストコンピュータ3000と通信が可能となっており、プリンタ1500内の情報等をホストコンピュータ30

00に通知可能に構成されている。

【0018】19はCPU12の主メモリ、ワークエリア等として機能するRAMであり、不図示の増設ポートに接続されるオプションRAMによりメモリ容量を拡張することができるように構成されている。尚、RAM19は出力情報展開領域、環境データ格納領域、NVRAM等に用いられる。上述したハードディスク(HD)、ICカード等の外部メモリ14は、メモリコントローラ(MC)20によってアクセスが制御される。また、外部メモリ14は、オプションとして接続され、フォントデータ、エミュレーションプログラム、フォームデータ等を記憶する。1501は操作パネルであり、操作のための各種スイッチ及びLED表示器等が配されている。

【0019】尚、上述した外部メモリは1個に限らず、少なくとも1個以上備え、内蔵フォントに加えてオプションフォントカード、言語系の異なるプリンタ制御言語を解釈するプログラムを格納した外部メモリを複数接続できるように構成されていても良い。更に、不図示のNVRAMを有し、操作パネル1501からのプリンタモード設定情報を記憶するようにしても良い。

【0020】図2は、プリンタ等の印刷装置が直接接続されているか、或いはネットワーク経由で接続されているホストコンピュータにおける典型的な印刷処理を実行するプログラムモジュールの構成を示す図である。ここで、アプリケーション201、グラフィックエンジン202、プリンタドライバ203及びシステムスプーラ204は、外部メモリ11に保存されたファイルとして存在し、実行される場合にOSやそのモジュールを利用する他のモジュールによってRAM2にロードされ、実行されるプログラムモジュールである。尚、アプリケーション201及びプリンタドライバ203は外部メモリ11のFDや不図示のCD-ROM或いは不図示のネットワークを経由して外部のハードディスク(HD)に追加することが可能となっている。

【0021】また、外部メモリ11に保存されているアプリケーション201はRAM2にロードされて実行されるが、このアプリケーション201からプリンタ1500に対して印刷を行う際には、同様にRAM2にロードされ実行可能となっているグラフィックエンジン202を利用して出力を行う。このグラフィックエンジン202は印刷装置毎に用意されたプリンタドライバ203と同様に、外部メモリ11からRAM2にロードされ、アプリケーション210からの出力をプリンタドライバ203を用いてプリンタの制御コマンドに変換する。プリンタドライバにより生成されたプリンタ制御コマンドはOSによってRAM2にロードされたシステムスプーラ204を経て、インタフェース21経由でプリンタ1500へ出力される仕組みとなっている。

【0022】図3は、ユーザがホストコンピュータ3000のアプリケーション201上から印刷を指示した場

合の印刷シーケンスを示すフローチャートである。

【0023】まず、印刷開始時に、アプリケーション201はグラフィックエンジン202に対して論理的な描画要求を行う(ステップS301)。この論理描画要求は、グラフィックエンジン202からプリンタドライバ203に伝えられる(ステップS302)。そして、プリンタドライバ203がこの論理描画要求を解釈し、プリンタが認識するプリンタ制御コマンドを生成し(ステップS303)、システムスプーラ204に送信する(ステップS304)。これにより、システムスプーラ204からプリンタ制御コマンドが適当なタイミングでプリンタへと送信され、プリンタで印刷がなされる。

【0024】図4は、本実施形態におけるプリンタドライバ203の構成を示すブロック図である。同図において、401は共通部であり、グラフィックエンジン202とのインタフェースが実装され、プリンタドライバ全体を制御する。402はUI部であり、プリンタドライバ機能をユーザが使うための設定を行うユーザインタフェースを出力する。403はサブモジュール群であり、UI部402のユーザインタフェースでユーザにより設定された印刷機能、例えば透かし印刷、ページ印刷などのドライバ付加機能に応じてロードされ、個々の機能を提供するために分割されたモジュールである。

【0025】ここで、プリンタドライバ203において行われる処理を図5及び図6に示すフローチャートを用いて説明する。

【0026】図5は、アプリケーション201からの印刷開始時に実行されるドライバモジュールの初期化処理を示すフローチャートである。アプリケーション201からの印刷開始指示により、プリンタドライバ203は初期化処理として、予めユーザが設定したプリンタドライバユーザインタフェース上のユーザ設定を検索する(ステップS501)。次に、プリンタドライバ203がサブモジュール群403としてN個の別々の機能を提供するモジュールを持っているとすれば、検索したユーザ設定の内容を参照してどのモジュールが必要か否かを判断し、必要なモジュールを外部メモリ11からロードする。

【0027】即ち、図5に示すように、UI部402が提供するユーザインタフェースからユーザの行ったプリンタドライバ設定が、モジュール1の機能を使用するならば(ステップS502のYes)、モジュール1をロードする(ステップS503)。また、ユーザの行ったプリンタドライバ設定が、モジュール2の機能を使用するならば(ステップS504のYes)、モジュール2をロードする(ステップS505)。同様に、ユーザの行ったプリンタドライバ設定が、モジュールNの機能を使用するならば(ステップS506のYes)、モジュールNをロードする(ステップS507)。これにより、初期化処理終了時、プリンタドライバ203には、

ユーザインタフェース上のユーザ設定の内容から必要なモジュールだけがロードされることになる。

【0028】次に、プリンタドライバ203の初期化処理が終了すると、プリンタドライバ203はグラフィックエンジン202を介して、アプリケーション201からの論理描画要求を受け付ける。

【0029】図6は、アプリケーション201からの論理描画要求を受け付けた時の処理を示すフローチャートである。まず、プリンタドライバ203がアプリケーション201からグラフィックエンジン202を介して伝えられた論理描画要求を共通部401を介して受け付ける（ステップS601）。

【0030】今、図5に示したプリンタドライバ初期化時に、モジュールk、モジュールl、…、モジュールmがロードされたとすれば、ある論理描画要求を受けた時に、その論理描画要求からプリンタが認識するプリンタ制御コマンドを生成するためにモジュールkを使用するか否かを決定する（ステップS602）。ここで、その論理描画要求を処理する際にモジュールkを使う必要があるならば（ステップS602のYes）、モジュールkを呼び出す（ステップS603）。そして、モジュールkは論理描画要求に対してプリンタドライバ特定の機能を実現する（ステップS604）。

【0031】上述した処理を全てのロードされたモジュール群に関して繰り返す（ステップS605～S610）。そして、実際に各サブモジュール群によって処理された論理描画要求はプリンタドライバ203の共通部401でプリンタ制御コマンドに変換される（ステップS611）。

【0032】この論理描画処理は描画終了まで繰り返され（ステップS612のYes）、共通部401で生成されたプリンタ制御コマンドは逐次システムスプーラ204に送信される（ステップS613）。

【0033】次に、複数のモジュールから構成されるプリンタドライバ203でロードするモジュールを決定する具体的な処理について説明する。

【0034】〔第1の実施形態〕第1の実施形態では、「透かし印刷機能」をユーザインタフェースにより設定（UI設定）する場合について説明する。

【0035】まず、印刷の前段階として、ユーザがプリンタドライバ203のユーザインタフェースを介してプリンタドライバ203の印刷設定を行う。ここでは、簡潔化のために、プリンタドライバ203は図7に示すようなユーザインタフェースをCRT10に表示し、ユーザにプリンタドライバ203の設定を行わせるものとする。

【0036】図7において、701は透かし印刷（全てのページに特定の文字を印刷する）機能をユーザがONに設定した状態を示し、また702はユーザが全てのページに「Confidential」の文字を透かし印刷することを

設定した状態を示している。尚、第1の実施形態ではプリンタドライバ203はモジュール1とモジュール2の2つのロード可能モジュールを持っているものとする。

【0037】ここで、UI設定による透かし印刷機能はモジュール1で使用する機能によりプリンタドライバ203内部で実装されているものとする。また、モジュール1はページの最後でそのページ上に特定の文字を印刷する機能を持つモジュールであるとする。

【0038】また、アプリケーション201は図8に示すような1ページデータを印刷出力するものとする。そして、図8に示すデータを印刷するためのアプリケーション201からの論理描画要求の例として図9に示すようなものを想定する。

【0039】図9に示す901のStartPage()はページの最初を示す論理描画要求である。また、902のWritePage()は1ページ目の座標10、10に“Page 1”を出力することを示す論理描画要求である。そして、903のEndPage()はページが終了したことを示す論理描画要求である。

【0040】図10は、第1の実施形態における初期化処理を示すフローチャートである。まず、印刷開始時、プリンタドライバ203はユーザインタフェースからユーザ設定を検索する（ステップS1001）。この際、図7に示すようにモジュール1を使用する「透かし印刷機能」が設定されているので（ステップS1002のYes）、モジュール1をロードする（ステップS1003）。また、モジュール2を使用するUI設定は行われていないので（ステップS1004のNo）、モジュール2はロードしない。

【0041】次に、アプリケーション201はグラフィックエンジン202に対してプリンタへの描画要求を繰り返し生成する。

【0042】図11は、第1の実施形態における論理描画処理を示すフローチャートである。尚、アプリケーション201からの論理描画要求はグラフィックエンジン202を介してプリンタドライバ203へと伝えられる。この時、アプリケーション201が行う論理描画要求は図9に示すようになり、901、902、903の順に論理描画要求がなされる。

【0043】まず、プリンタドライバ203がグラフィックエンジン202からStartPage描画要求901を受け（ステップS1101）、この論理描画要求はモジュール1を必要としないため（ステップS1102のNo）、そのまま共通部401でプリンタ制御コマンドに変換される（ステップS1105）。そして、変換されたプリンタ制御コマンドはシステムスプーラ204へと送信される（ステップS1107）。未だ描画途中であるため（ステップS1106のNo）、プリンタドライバ203は再びステップS1101に戻り、グラフィックエンジン202より論理描画要求としてWritePageを

受け付ける。

【0044】次に、プリンタドライバ203がグラフィックエンジン202から受け付けたWritePage描画要求902はモジュール1を必要としないため（ステップS1102のNo）、そのまま共通部401でプリンタ制御コマンドに変換され（ステップS1105）、システムスプーラ204へと送信される（ステップS1107）。未だ描画途中であるため（ステップS1106のNo）、プリンタドライバ203は再びステップS1101に戻り、グラフィックエンジン202より論理描画要求としてEndPageを受け取る。この論理描画要求はページの終りを示し、ページの最後で特定の文字を印刷する透かし印刷機能を使用するので（ステップS1102のYes）、モジュール1が呼び出される（ステップS1103）。このモジュール1にて透かし印刷機能処理がなされ（ステップS1104）、共通部401でプリンタ制御コマンドに変換され（ステップS1105）、システムスプーラ204へと送られる（ステップS1107）。

【0045】図12は、透かし印刷機能を実行するプリンタ制御コマンドの一例を示す図である。図12に示す1201のWriteWaterMark（）は、ページの中央部に文字列「Confidential」を付加することを意味する論理描画である。

【0046】以上の処理により、第1の実施形態における印刷要求が終了する（ステップS1106のYes）。

【0047】図13は、第1の実施形態における印字結果を示す図である。図示するように、ページの中央部に文字列「Confidential」1301が透かし印刷される。

【0048】以上説明した第1の実施形態によれば、ユーザによるユーザインタフェース上の設定からプリンタドライバにロードするモジュール群を動的に決定することにより、ユーザによる設定上、必要としない冗長部分をプリンタドライバにロードしないこととなり、モジュール全体のファイルサイズを減少させることが可能となる。これは結果としてホストコンピュータのメモリ不足を解決し、冗長部分をロードしないことにより、処理が簡素化され印刷スループットを向上させることが可能となる。

【0049】〔第2の実施形態〕次に、図面を参照しながら本発明に係る第2の実施形態について説明する。

【0050】第2の実施形態では、第1の実施形態の「透かし印刷機能」に加えてユーザが「ページ番号印刷機能」をUI設定する場合について説明する。

【0051】まず、印刷の前段階として、ユーザがプリンタドライバ203のユーザインタフェースを介してプリンタドライバ203の印刷設定を行う。ここでは、簡潔化のために、プリンタドライバ203は図14に示すようなユーザインタフェースをCRT10に表示し、ユ

ーザにプリンタドライバ203の設定を行わせるものとする。

【0052】図14において、1401は透かし印刷（全てのページに特定の文字を印刷する）機能をユーザがONに設定した状態を示し、また1402はユーザが全てのページに「Confidential」の文字を透かし印刷することを設定した状態を示している。更に、1403はページ番号印刷（ページ番号を印刷する）機能をユーザが設定した状態を示している。尚、第2の実施形態ではプリンタドライバ203はモジュール1とモジュール2の2つのロード可能モジュールを持っているものとする。

【0053】ここで、UI設定による透かし印刷機能はモジュール1で使用する機能によりプリンタドライバ203内部で実装されているものとする。また、モジュール1はページの最後にそのページ上に特定の文字を印刷する機能をもつモジュールであるとする。また、モジュール2はページの最後にそのページ上にページ番号を印刷する機能をもつモジュールであるとする。

【0054】また、アプリケーション201は第1の実施形態と同様に、図8に示すような1ページデータを印刷出力するものとする。そして、図8に示すデータを印刷するためのアプリケーション201からの論理描画要求の例として図9に示すようなものを想定する。

【0055】図9に示す901のStartPage（）はページの最初を示す論理描画要求である。また、902のWritePage（）は1ページ目の座標10、10に「Page 1」を出力することを示す論理描画要求である。そして、903のEndPage（）はページが終了したことを示す論理描画要求である。

【0056】ここで、第2の実施形態における初期化処理は前述した第1の実施形態で説明した図10と同様であり、図10を参照しながら説明する。

【0057】まず、印刷開始時、プリンタドライバ203はユーザインタフェースからユーザ設定を検索する（ステップS1001）。この際、図14に示すようにモジュール1を使用する透かし印刷機能が設定されているので（ステップS1002のYes）、モジュール1をロードする（ステップS1003）。また、図14に示すようにモジュール2を使用する「ページ番号印刷機能」が設定されているので（ステップS1004のYes）、モジュール2をロードする（ステップS1005）。

【0058】次に、アプリケーション201はグラフィックエンジン202に対してプリンタへの描画要求を繰り返し生成する。

【0059】図15は、第2の実施形態における論理描画処理を示すフローチャートである。尚、アプリケーション201からの論理描画要求はグラフィックエンジン202を介してプリンタドライバ203へと伝えられ

る。この時、アプリケーション201が行う論理描画要求は図9に示すようになり、901、902、903の順に論理描画要求がなされる。

【0060】まず、プリンタドライバ203がグラフィックエンジン202からStartPage描画要求901を受け(ステップS1501)、この論理描画要求はモジュール1を必要としないため(ステップS1502のNo)、モジュール1は呼び出されない。また、この論理描画要求はモジュール2も必要としないため(ステップS1505のNo)、この論理描画要求は、そのまま共通部401でプリンタ制御コマンドに変換される(ステップS1508)。そして、変換されたプリンタ制御コマンドはシステムスプーラ204へと送信される(ステップS1510)。この時、描画途中であるため(ステップS1509のNo)、プリンタドライバ203は再びステップS1501に戻り、論理描画要求を受け付ける。

【0061】次に、プリンタドライバ203がグラフィックエンジン202より受け付けたWritePage描画要求902はモジュール1を必要としないため(ステップS1502のNo)、モジュール1を呼び出されない。また、この論理描画要求はモジュール2も必要としないため(ステップS1505のNo)、この論理描画要求は、そのまま共通部401にてプリンタ制御コマンドに変換される(ステップS1508)。その後、プリンタ制御コマンドはシステムスプーラ204へと送信される(ステップS1510)。この時、描画途中であるため(ステップS1509のNo)、プリンタドライバ203は再びステップS1501に戻り、論理描画要求を受け付ける。

【0062】次に、プリンタドライバ203はグラフィックエンジン202より論理描画要求としてEndPageを受け取る。この論理描画要求はページの終りを示し、ページの最後で特定の文字を印刷する透かし印刷機能を使用するので(ステップS1502のYes)、モジュール1が呼び出される(ステップS1503)。このモジュール1にて透かし印刷機能処理がなされる(ステップS1504)。また、この論理描画要求はページの終りを示し、ページの最後でページ番号を印字するページ番号印刷機能を使用するので(ステップS1505のYes)、モジュール2が呼び出される(ステップS1506)。このモジュール2にてページ番号印刷機能処理がなされる(ステップS1507)。そして、共通部401でプリンタ制御コマンドに変換され(ステップS1508)、システムスプーラ204へ送信される(ステップS1510)。

【0063】図16は、透かし印刷機能とページ印刷機能を実行するプリンタ制御コマンドの一例を示す図である。図16に示す1601のWriteWaterMark()は、ページの中央部に文字列「Confidential」を付加すること

を意味する論理描画である。また、1602のWritePageNumber()は、ページ番号を印刷することを意味する論理描画である。

【0064】以上の処理により、第2の実施形態における印刷要求が終了する(ステップS1509のYes)。

【0065】図17は、第2の実施形態における印字結果を示す図である。図示するように、ページの中央部に文字列「Confidential」1701が透かし印刷され、ページ番号1702が印刷される。

【0066】以上説明した第2の実施形態によれば、尚、本発明は複数の機器(例えば、ホストコンピュータ、インタフェイス機器、リーダー、プリンタなど)から構成されるシステムに適用しても、一つの機器からなる装置(例えば、複写機、ファクシミリ装置など)に適用してもよい。

【0067】また、本発明の目的は前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システム或いは装置に供給し、そのシステム或いは装置のコンピュータ(CPU若しくはMPU)が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。

【0068】この場合、記憶媒体から読出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

【0069】プログラムコードを供給するための記憶媒体としては、例えばフロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモ리카ード、ROMなどを用いることができる。

【0070】また、コンピュータが読出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS(オペレーティングシステム)などが実際の処理の一部又は全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0071】更に、記憶媒体から読出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部又は全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0072】

【発明の効果】以上説明したように、本発明によれば、ユーザが設定した印刷機能に応じて、出力機器を制御す

るドライバモジュールに組み込むモジュール群を決定することにより、冗長性を減少させ、メモリ不足を軽減でき、印刷処理を簡潔化し、印刷速度を向上させることが可能となる。

【図面の簡単な説明】

【図1】本実施形態による印刷システムの構成を示すブロック図である。

【図2】ホストコンピュータにおける典型的な印刷処理を実行するプログラムモジュールの構成を示す図である。

【図3】ユーザがホストコンピュータ3000のアプリケーション201上から印刷を指示した場合の印刷シーケンスを示すフローチャートである。

【図4】本実施形態によるプリンタドライバ203の構成を示すブロック図である。

【図5】アプリケーション201からの印刷開始時に実行されるドライバモジュールの初期化処理を示すフローチャートである。

【図6】アプリケーション201からの論理描画要求を受け付けた時の処理を示すフローチャートである。

【図7】第1の実施形態におけるプリンタドライバのユーザインタフェースを示す図である。

【図8】アプリケーション201上から印刷出力するデータを示す図である。

【図9】アプリケーション201からの論理描画要求の例を示す図である。

【図10】第1の実施形態における初期化処理を示すフローチャートである。

【図11】第1の実施形態における論理描画処理を示すフローチャートである。

【図12】透かし印刷機能を実行するプリンタ制御コマンドの一例を示す図である。

【図13】第1の実施形態における印字結果を示す図である。

【図14】第2の実施形態におけるプリンタドライバのユーザインタフェースを示す図である。

【図15】第2の実施形態における論理描画処理を示すフローチャートである。

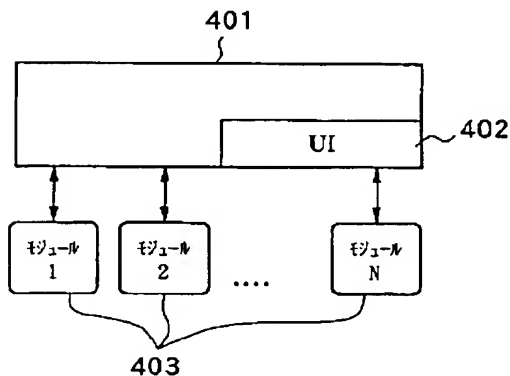
【図16】透かし印刷機能とページ印刷機能を実行するプリンタ制御コマンドの一例を示す図である。

【図17】第2の実施形態における印字結果を示す図である。

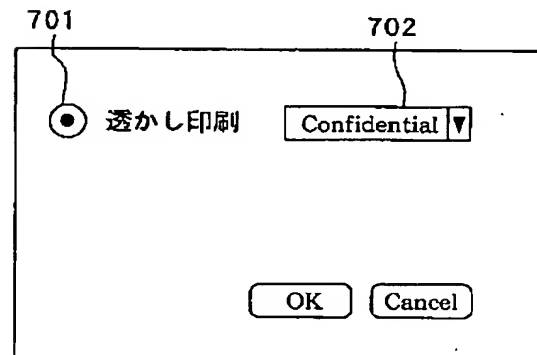
【符号の説明】

- 1 CPU
- 2 RAM
- 3 ROM
- 4 システムバス
- 12 CPU
- 13 ROM
- 19 RAM
- 201 アプリケーション
- 202 グラフィックエンジン
- 203 プリンタドライバ
- 204 システムスプーラ
- 401 共通部
- 402 UI部
- 403 サブモジュール群
- 1500 プリンタ
- 3000 ホストコンピュータ

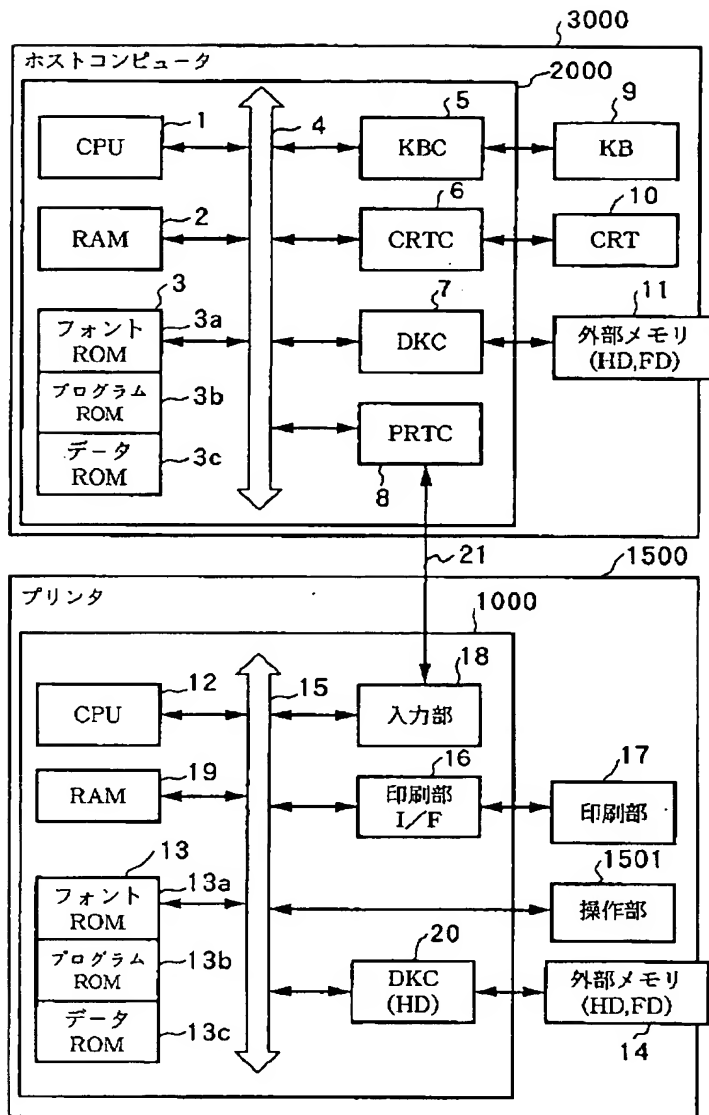
【図4】



【図7】



【図1】



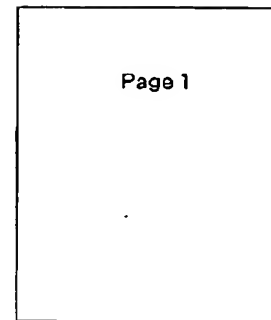
【図9】

```

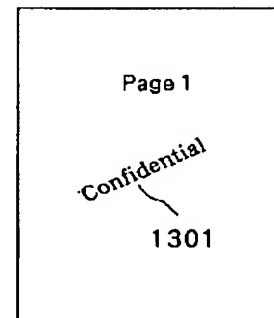
901      902      1201
StartPage() {      WriteWaterMark("Confidential")
WritePage (1 , 10 , 10 , "Page1") ; EndPage()
EndPage()
          {
903

```

【図8】

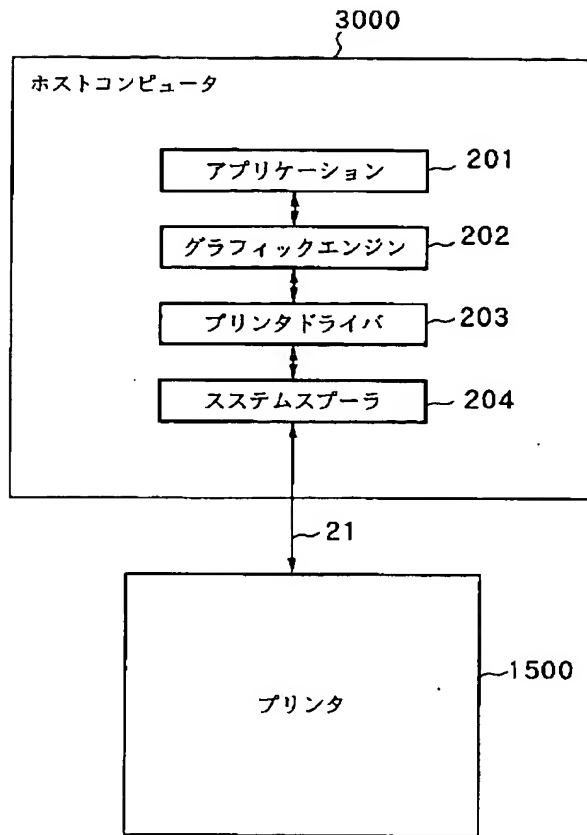


【図13】

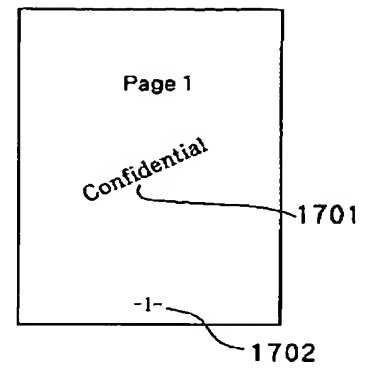


【図12】

【図2】



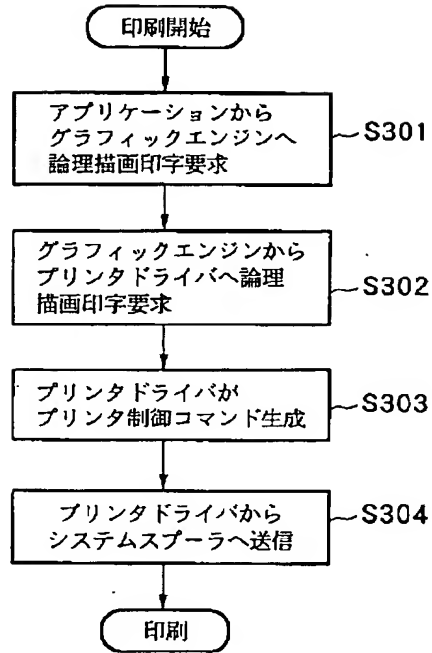
【図17】



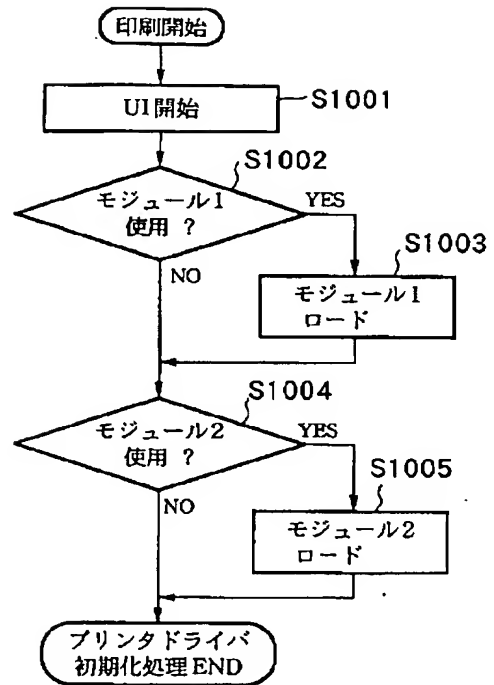
【図16】

```
1601
{
WriteWaterMark("Confidential")
WritePageNumber() 1602
EndPage()
}
```

【図3】



【図10】



【図14】

1401

1402

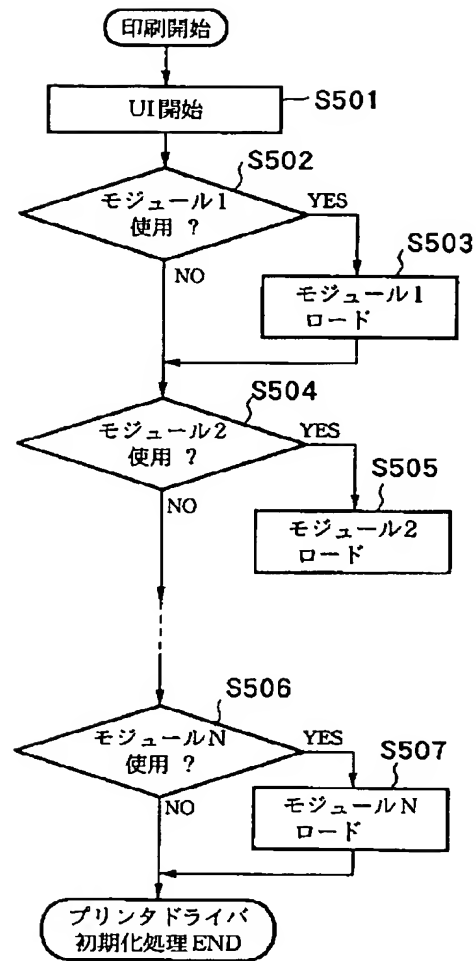
☒ 透かし印刷 Confidential ▼

☒ ページ番号印刷

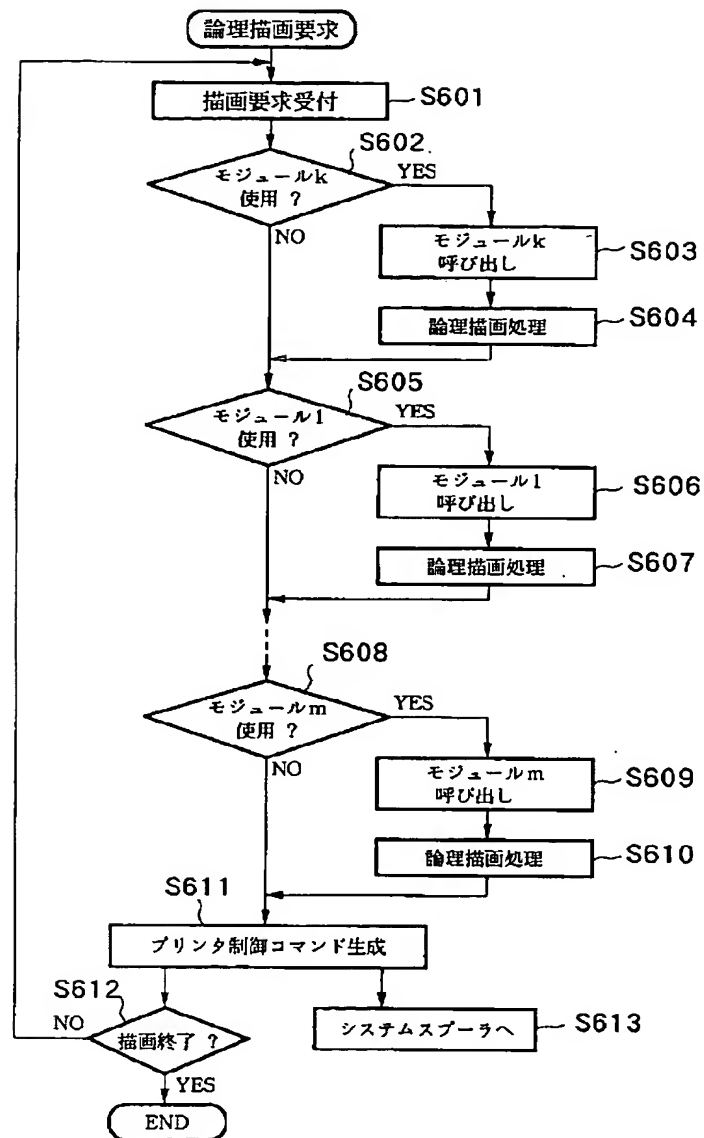
1403

OK Cancel

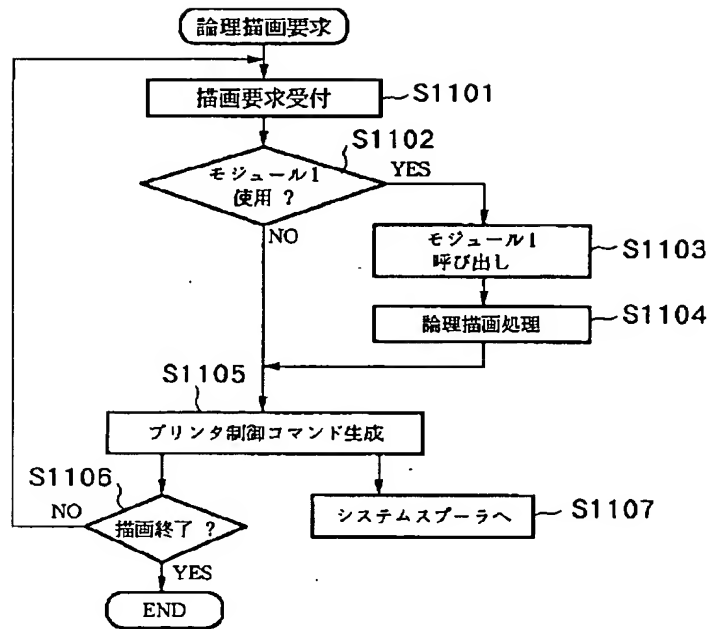
【図5】



【図6】



【図11】



【図15】

